JBoss Operations Network 3.2
3.2 Release Notes

---

for release information
Edition 3.2

Ella Deon Ballard

JBoss Operations Network 3.2 3.2 Release Notes

for release information
Edition 3.2

Ella Deon Ballard
dlackey@redhat.com

Legal Notice

Copyright © 2013 Red Hat, Inc..

This document is licensed by Red Hat under the Creative Commons Attribution-ShareAlike 3.0 Unported License. If you distribute this document, or a modified version of it, you must provide attribution to Red Hat, Inc. and provide a link to the original. If the document is modified, all Red Hat trademarks must be removed.

Red Hat, as the licensor of this document, waives the right to enforce, and agrees not to assert, Section 4d of CC-BY-SA to the fullest extent permitted by applicable law.

Red Hat, Red Hat Enterprise Linux, the Shadowman logo, JBoss, MetaMatrix, Fedora, the Infinity Logo, and RHCE are trademarks of Red Hat, Inc., registered in the United States and other countries.

Linux ® is the registered trademark of Linus Torvalds in the United States and other countries.

Java ® is a registered trademark of Oracle and/or its affiliates.

XFS ® is a trademark of Silicon Graphics International Corp. or its subsidiaries in the United States and/or other countries.

MySQL ® is a registered trademark of MySQL AB in the United States, the European Union and other countries.

Node.js ® is an official trademark of Joyent. Red Hat Software Collections is not formally related to or endorsed by the official Joyent Node.js open source or commercial project.

The OpenStack ® Word Mark and OpenStack Logo are either registered trademarks/service marks or trademarks/service marks of the OpenStack Foundation, in the United States and other countries and are used with the OpenStack Foundation's permission. We are not affiliated with, endorsed or sponsored by the OpenStack Foundation, or the OpenStack community.

All other trademarks are the property of their respective owners.

Abstract

These release notes contain important information about new features, known issues, and other technical notes available at the time that JBoss Operations Network 3.2 was released.

Table of Contents

1. New Features in JBoss Operations Network 3.2 2
1.1. Updated: New Installation Script and Simplified Installation Process 2
1.2. Updated: New High-Speed Metric Data Storage 2
1.3. Updated: New Control Script for Server Management 3
1.4. New: Fine-grained Permissions for Bundle Management 3
1.5. Enhanced: Creating Subgroups from Dynagroups 4
1.6. New: Dynagroup Methods in the Remote API 4
1.7. New: Authenticating to an SVN Repository to Upload Bundles 4
1.8. New: Additional JBoss Enterprise Application Platform 6 Metrics 4
1.9. Updated: New Chart and Metric Graph Designs 5
1.10. New: Remove an Agent through the UI 7
1.11. New: Editing Alert Conditions 7
1.12. New: Scheduling Operations Using cron Jobs 7
1.13. New: Specifying Alternate Trap OIDs for SNMP Alerts 7
1.14. New: Ignoring Resource Types 7
1.15. New: Synchronous Availability Check 7
1.16. New: Including Group Operations in Portlets 8
1.17. Full Support: REST API 8
1.18. New: Page Controls for LDAP Group Operations 8

2. Structural Changes in JBoss ON 3.2 8
2.1. Bundle Permissions 8
2.2. New Start Options for the Agent 9
2.3. New Attribute in the Plug-in Descriptor 9
2.4. Remote API Changes 9
2.5. Discovery Callbacks for Agent Plug-ins 10

3. JBoss Operations Network Prerequisites and Basic Installation 10
3.1. Supported Configuration Information 10
3.2. Quick Setup for JBoss Operations Network 3.2 11
3.3. Upgrading to JBoss Operations Network 3.2 12

4. Known Issues in JBoss Operations Network 14
4.1. Installation and Upgrade 14
4.2. Windows 15
4.3. Agents 15
4.4. Server 16
4.5. Platform Plug-in 17
4.6. EAP 6 Plug-in 17
4.7. EAP 5 Plug-in 18
4.8. EAP 4 Plug-in 19
4.9. UI 19
4.10. Provisioning 19
4.11. Monitoring 20

5. Document History 20

These release notes contain feature information, changed support, and structural changes in JBoss Operations Network 3.2.

JBoss Operations Network provides centralized control, configuration, and visibility to JBoss applications and related infrastructure. Broad and detailed monitoring, versioned application deployment, auditable configuration management, and controlled software patching through JBoss ON reduce the complexity, improve the reliability, and decrease the cost of large scale JBoss deployments.

This version of JBoss Operations Network contains new features, enhancements, and bug fixes. It is recommended that all JBoss Operations Network users upgrade to JBoss Operations Network 3.2.

1. New Features in JBoss Operations Network 3.2

This version of JBoss ON introduces both new features and feature enhancements that improve JBoss ON's performance for managing resources.

1.1. Updated: New Installation Script and Simplified Installation Process

The overall server installation process has been streamlined in JBoss ON 3.2 as part of upgrading the underlying application server to JBoss Enterprise Application Platform 6.

The biggest single change is that there is no longer a web-based configuration process. All configuration is performed by the installation script, based on the values in the **`rhq-server.properties`** file. This makes it much easier to automate server provisioning.

There are other changes that simplify the installation process:

- There is a new installation and upgrade script, **`rhqctl install`** and **`rhqctl upgrade`**.
- By default, multiple components (the server, the agent on the server platform, and the metrics storage node) are installed with the installation script. These components can also be installed individually, using the appropriate options (**`--server`**, **`--storage`**, **`--agent`**).
- The web installer has been removed and the entire process is configured through the command-line scripts and property files.
- The **`rhq-server.properties`** file contains the default configuration for a local PostgreSQL server. If an Oracle server is used, then the properties file should be edited before running the installation script to point to the database.

1.2. Updated: New High-Speed Metric Data Storage

Starting in JBoss ON 3.2, all numeric monitoring data and aggregated values are stored in a dedicated NoSQL database, the JBoss ON *metrics storage node*. Every server is either installed (or upgraded with) a metrics storage node or must point to an existing storage node, in addition to the SQL database used to store all other JBoss ON resource data and configuration.

NOTE

The data stored in the metrics storage node is the numeric or metric data; this information is collected and updated frequently.

Other types of monitoring information — such as calltime metrics and traits — are still stored in the relational database.

There are a number of performance benefits by splitting the monitoring data into a separate storage node. This improves the performance of the backend SQL database by moving heavy write traffic to a separate database, and it also frees resources for intensive features like drift snapshots and stored bundles and content. Additionally, the NoSQL database cluster can be easily expanded to include additional nodes, even after server installation. This reduces the structural database bottleneck that has historically limited metric collection levels.

The metric data storage changes primarily affect the installation process:

- A storage node must be installed with the server or referenced by the server properties. This is handled through the **`rhqctl`** setup script, by default.
- Additional nodes can be added (or removed) on additional hardware, also using the **`rhqctl`** script.
- Every metrics storage node requires a local agent, even if no server is installed.
- The **`Administration > System Settings`** page has historically had rules about how long to store monitoring and event data. *This configuration remains the same*. The same data storage configuration is still configured on that page; it simply applies to the metrics storage node rather than the external SQL database.

There is an option with the **`rhqctl upgrade`** script to migrate all existing monitoring data out of the SQL database and into the new metrics storage database. If that option (**`--run-data-migrator`**) is not used with the upgrade, then the old monitoring data are lost.

1.3. Updated: New Control Script for Server Management

Previous versions of JBoss ON used the **`rhq-server.sh|bat`** script to install and start the server. This has been replaced by the new **`rhqctl`** script. This script has specific commands to install and upgrade the server, to install and upgrade associated components such as the agent and a metrics storage node, and to start and stop the server, agent, and storage node.

This new **`rhqctl`** control script still uses the **`rhq-server.properties`** configuration file for its initial configuration.

1.4. New: Fine-grained Permissions for Bundle Management

In previous versions of JBoss ON, there was a single (extremely powerful) permission for bundle operations: **`MANAGE_BUNDLES`**. This allowed complete access to everything related to the bundle configuration: resources, groups, bundle content and configuration.

In JBoss ON 3.2, the bundle permissions have been broken into multiple permissions, at three levels:

- Global permissions to manage bundle groups, deploy bundles, view bundles, create bundles, and delete bundles.

- Bundle group-level permissions to manage the bundle groups by assigning and unassigning existing bundles to the group and by creating and deleting new bundles for that group.
- Resource group-level permissions to allow a user to deploy a bundle to a resource group over which he has management permissions.

This new permissions structure introduces the idea of a bundle group. A resource group is a defined group of resources and access controls can be set based on what resource group a user has permissions to. Likewise, a bundle group is a defined subset of existing bundles, and a user is granted access to a specified group. A user can only view and manage bundles which belong to that given bundle group.

These changes allow much better control over content streams and resource maintenance, while providing visibility (without power) to managers and analysts who require clarity about development and production content.

1.5. Enhanced: Creating Subgroups from Dynagroups

Dynagroups are dynamic groups that are created through a versatile expression language which filters members on criteria such as name, parent or child resources, resource type, and version.

Previously, a dynagroup expression evaluated the entire inventory when determining group membership. A new criterion has been added, **`memberof`**, which means that members of the dynagroup are evaluated from the members of an existing group. In other words, the dynagroup is a subgroup or subset of an existing group.

This is especially useful in creating fine-grained access controls and other management operations.

1.6. New: Dynagroup Methods in the Remote API

Previously, the classes and methods to create dynagroups were internal. Therefore, dynagroup expressions could only be created and edited through The JBoss ON GUI. The updated remote API (used by JBoss ON server-side scripts for alerts, the JBoss ON CLI, and other clients) now includes the full **`GroupDefinitionManager`** to create, edit, and delete dynamic groups.

1.7. New: Authenticating to an SVN Repository to Upload Bundles

Previously, there were limits on sites where JBoss ON could download bundlefiles; only the URL could be specified, no authentication credentials. In practice, this meant that most commonly bundles were uplooaded from a local system to JBoss ON.

Starting in JBoss ON 3.2, JBoss ON can pass authentication credentials to any remote site that is specified. A bundle definition can be created that references the SVN repository URL and then authenticates to the SVN repository (either through standard HTTP or over HTTPS). From there, it pulls in the bundle archive.

This makes it easier for bundles to be pulled directly from a build system.

1.8. New: Additional JBoss Enterprise Application Platform 6 Metrics

A number of new metrics related to session management and web application responsiveness have been added to the JBoss Enterprise Application Platform 6 plug-in for deployed web context (WAR) resources. These include:

- The virtual host with which this context is associated

- Duplicated session IDs
- Currently active sessions
- Maximum Active Sessions
- Created Sessions
- Created Sessions per Minute
- Expired Sessions
- Expired Sessions per Minute
- Rejected Sessions
- Rejected Sessions per Minute
- Average Session Alive Time
- Max Session Alive Time

1.9. Updated: New Chart and Metric Graph Designs

The charts used to render the monitoring graphs have been updated.

- The **Monitoring** tab has been streamlined.

  The **Availability** tab, which was a table of availability changes, as been moved onto the new **Metrics** tab.

  Additionally, the **Tables** and **Graphs** subtabs have been combined into a single view on the **Metrics** tab, where each row in the table can be expanded to show the visual graph.

- It is now possible to click and drag on any point on a metrics graph and zoom into the specified time period, with updated calculations for that new window.
- The overall **Monitoring** tab has been reorganized so that monitoring data is first displayed in a table grid, and then each metric is expanded into a visual graph. This allows the details of metrics data to be more easily grasped.
- Stylistically, the availability chart has been streamlined into a clear linear graph, and metrics charts are bar graphs with bands of color to help show minimum and maximum readings per aggregated data point.

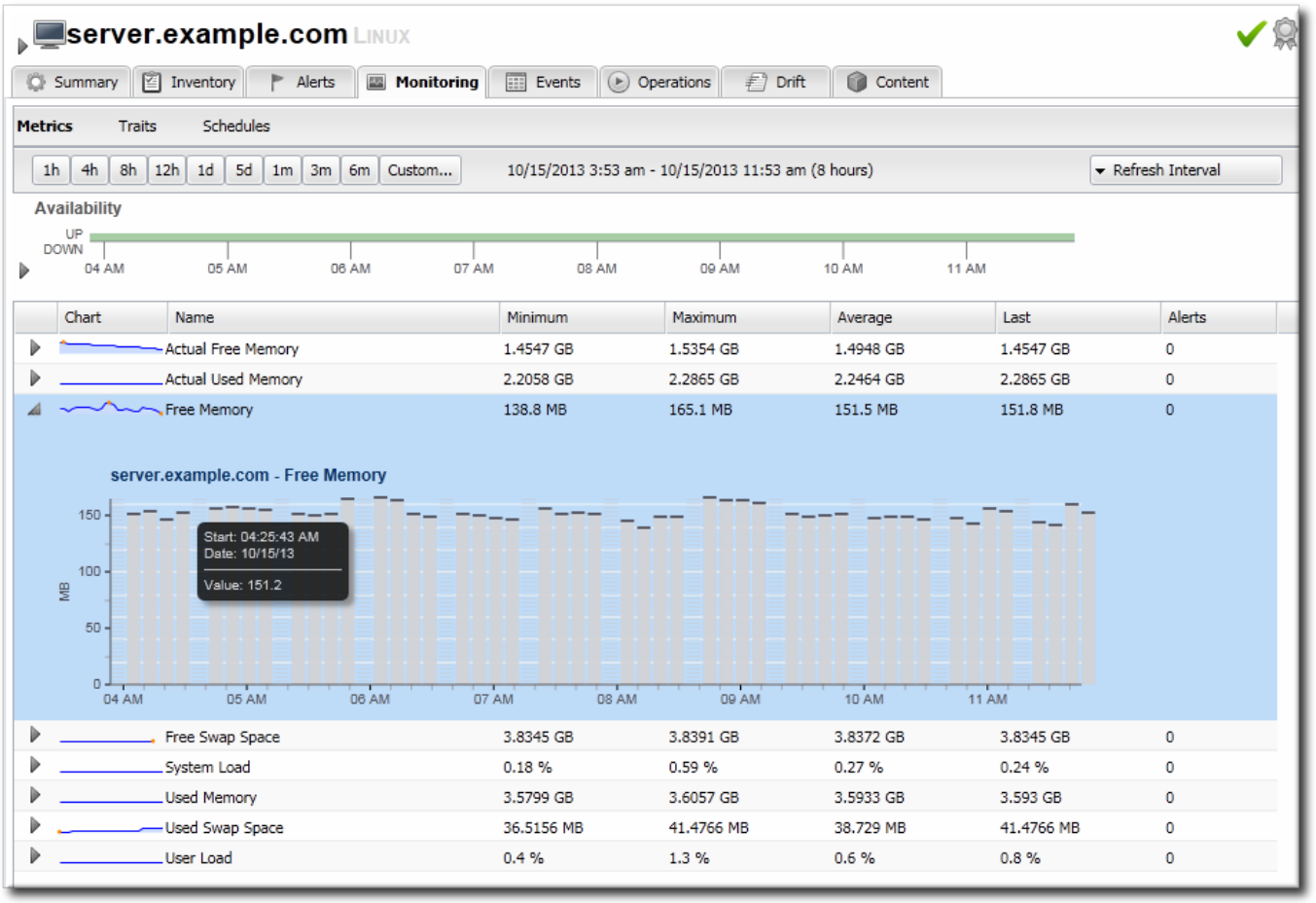

Figure 1. New Graph Design

- The graphs have been enhanced so that hovering over a data point gives information about the aggregated data value, the minimum and maximum readings for the time period, and the start/end times of the data collection period.
- Metrics can be selected on any resource **`Monitoring`** tab and sent to any saved dashboard.

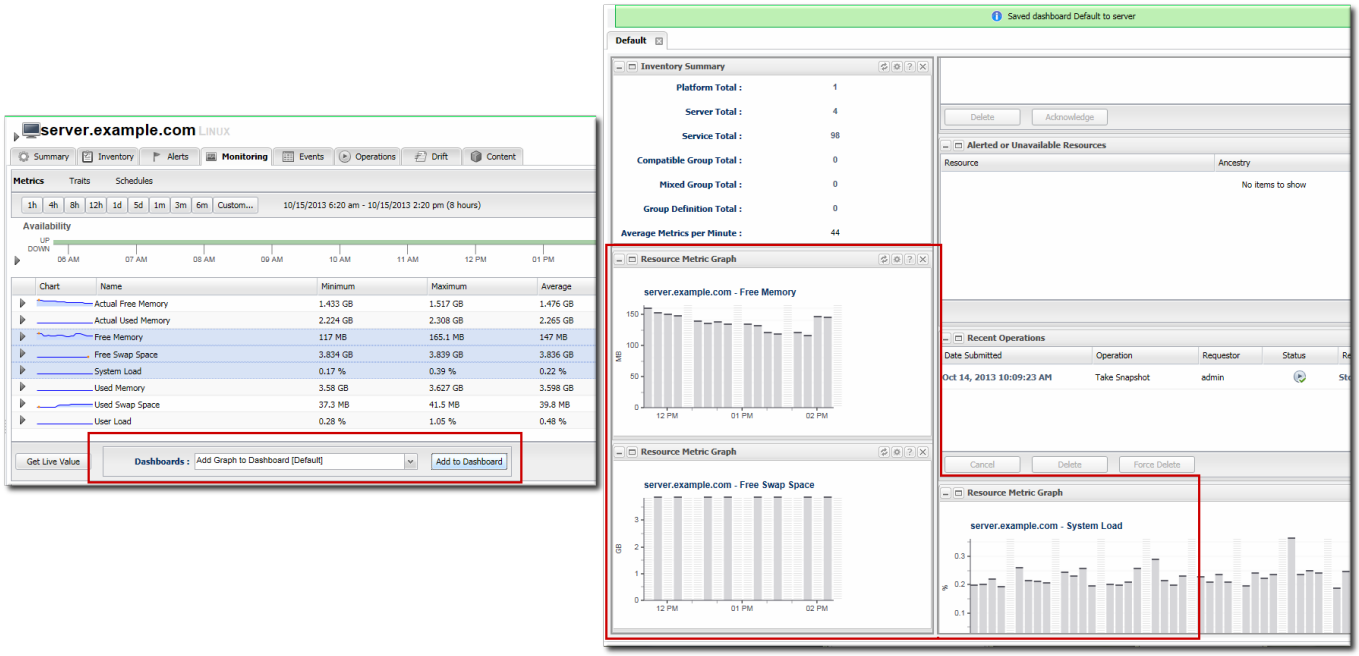

Figure 2. Sending Charts to the Dashboard

1.10. New: Remove an Agent through the UI

Previously, JBoss ON agents could not be removed through the JBoss ON web UI. They could only be removed from the host system, and then the platform entry had to be imported into the inventory (if it wasn't already) and then removed from the server's inventory. Removing the platform resource ultimately removed the agent's registration.

Now, there is an option in the **Administration > Agents** page to delete a selected agent. This removes the agent's registration information and its entire inventory directly, without having to go through a platform resource.

1.11. New: Editing Alert Conditions

Previously, an alert condition within an alert definition could not be edited. It had to be deleted and then replaced by a new condition.

Now, alert conditions can be edited within the definition.

1.12. New: Scheduling Operations Using cron Jobs

The JBoss ON web UI has long allowed adminitrators to schedule an operation using a cront string. However, it was not possible to use a cron string when creating operations in the CLI or scripts. The **OperationManagerRemote** remote API method has been updated to allow the schedule to be created in cron syntax.

1.13. New: Specifying Alternate Trap OIDs for SNMP Alerts

Previously, all SNMP alert notifications sent out the same trap OID, which was specified in the global configuration for the SNMP alert sender plug-in.

Starting with JBoss ON 3.2, it is possible to set a different OID within the alert definition. Now, the OID set in the global configuration is used as an enterprise identifier. Then, the SNMP trap OID set in the alert definition is used in a special attribute called a *variable binding* along with other identifying information for the alert.

1.14. New: Ignoring Resource Types

It has always been possible to ignore discovered resources in the discovery queue. Now, it is possible in the **Administration** tab to define resource types which will always be ignored and blocked from the inventory.

This is exceptionally useful for server types which have a large number of child resources. Child server and service resources are automatically imported when its parent is imported. For example, a JBoss Enterprise Application Platform resource may have hundreds of MBean child resources; ignoring that resource type can make managing the JBoss Enterprise Application Platform server easier and more maintainable.

1.15. New: Synchronous Availability Check

Resource availability is checked on a schedule (such as every minute for servers and every 10 minutes for services). This meant that there could be a gap between the collection time and the availability state displayed in the web UI or returned by API calls.

JBoss ON 3.2 changed availability checking so that it is performed every 15 seconds when a resource page is open or when a call is made about the resource. This allows for near-immediate visibility of when the state of a resource changes.

When a resource is not being viewed, the normal availability schedule applies.

1.16. New: Including Group Operations in Portlets

Previously, the **`Scheduled Operations`** portlet in the **`Dashboard`** listed operations scheduled on individual resources. Now, the list includes operations scheduled for compatible groups along with the resource operations.

1.17. Full Support: REST API

In JBoss ON 3.1, a new REST API was introduced as a tech preview feature.

Starting in JBoss ON 3.2, the REST API is fully supported. The REST API, along with some example applications, is available with the JBoss ON server:

```
https://server.example.com:7080/rest
```

Along with being supported, the REST API has been extended to include additional functionality, including:

- LDAP authentication support
- Paging on returns by following RFC 5988

1.18. New: Page Controls for LDAP Group Operations

With Active Directory and other LDAP directories, the membership lists returned with LDAP authorization could exceed the search limits set for the directory. This could lead to inconsistent or incomplete group information when working with LDAP groups in JBoss ON.

JBoss ON 3.2 implements LDAP paging controls for search results, which helps improve the performance for handling large groups by breaking the group results into pages that are within the search limits.

2. Structural Changes in JBoss ON 3.2

There are some structural changes between JBoss Operations Network 3.2 and 3.1 which are called out because they affect the usage and experience of JBoss ON.

2.1. Bundle Permissions

New, separate permissions and access control structures have been created for bundle management. This offers fine-grained control, with different permissions for different actions and a new concept of bundle groups, so access can be restricted to different content streams.

Previously, there was a single bundle-related permission, **`MANAGE_BUNDLES`**, which granted access to all bundles. This is an extremely powerful permission, since it grants unlimited rights over all bundle destinations and content. If any users were assigned this permission in a previous release, **that permission remains in effect**, without having to edit the roles.

2.2. New Start Options for the Agent

When configuring the agent to run as a service, the agent can be configured to start with a custom start command. This is used mainly to start the agent using **su** or **sudo**, allowing the agent to run as a different user.

The start command is defined with the other agent properties in the **rhq-agent-env.sh** file. There are two parts to the configuration: the start command itself and then an optional setting to enable a password prompt.

```
RHQ_AGENT_START_COMMAND="su -m test -c '${RHQ_AGENT_HOME}/bin/rhq-agent.sh'"
RHQ_AGENT_PASSWORD_PROMPT=true
```

Setting a start command overrides whatever command is passed in the command line to start the agent.

IMPORTANT

The password prompt should never be enabled for an agent on a server machine, which is managed by **rhqctl**. The **rhqctl** never directs input to the agent process, so it is not possible to input a password, and the start process will hang.

2.3. New Attribute in the Plug-in Descriptor

The **<c:option-source>** node has a new attribute, *expressionScope*, to help refine expression searches. This attribute links configuration properties to resources or configurations.

2.4. Remote API Changes

Table 1. Deprecated Methods

Deprecated Method	New or Replacement Method
org.rhq.core.util.exec.ProcessExecutor#redirectStreams	org.rhq.core.util.exec.ProcessExecutor#redirectAllStreams
org.rhq.bundle.ant.AntLauncher#AntLauncher()	org.rhq.bundle.ant.AntLauncher#AntLauncher(boolean requireExplicitCompliance)
org.rhq.bundle.ant.type.DeploymentUnitType#getManageRootDir()	org.rhq.bundle.ant.type.DeploymentUnitType#getCompliance()
org.rhq.bundle.ant.type.DeploymentUnitType#setManageRootDir(String booleanString)	org.rhq.bundle.ant.type.DeploymentUnitType#setCompliance(org.rhq.core.util.updater.DestinationComplianceMode)
org.rhq.core.util.updater.DeploymentData#DeploymentData(DeploymentProperties deploymentProps, Set zipFiles, Map rawFiles, File sourceDir, File destinationDir, Map zipEntriesToRealizeRegex, Set rawFilesToRealize, TemplateEngine templateEngine, Pattern ignoreRegex, boolean manageRootDir, Map zipsExploded)	org.rhq.core.util.updater.DeploymentData#DeploymentData(DeploymentProperties deploymentProps, Set zipFiles, Map rawFiles, File sourceDir, File destinationDir, Map zipEntriesToRealizeRegex, Set rawFilesToRealize, TemplateEngine templateEngine, Pattern ignoreRegex, Map zipsExploded)
The *managedRootDir* parameter in org.rhq.core.util.updater.DeploymentData#DeploymentData has been removed.	org.rhq.core.util.updater.DeploymentProperties#getDestinationCompliance()

Deprecated Method	New or Replacement Method
org.rhq.core.util.updater.DeploymentData#isManageRootDir()	org.rhq.core.util.updater.DeploymentProperties#getDestinationCompliance()
org.rhq.core.util.updater.DeploymentProperties#DeploymentProperties(int deploymentId, String bundleName, String bundleVersion, String description)	org.rhq.core.util.updater.DeploymentProperties#DeploymentProperties(int, String, String, String, DestinationComplianceMode)
org.rhq.core.util.updater.DeploymentProperties#getManageRootDir()	org.rhq.core.util.updater.DeploymentProperties#getDestinationCompliance()
org.rhq.core.util.updater.DeploymentProperties#setManageRootDir(boolean willManageRootDir)	org.rhq.core.util.updater.DeploymentProperties#setDestinationCompliance(DestinationComplianceMode mode)
org.rhq.core.util.exec.ProcessExecutor#redirectStreams()	org.rhq.core.util.exec.ProcessExecutor#redirectAllStreams()
org.rhq.enterprise.server.resource.group.LdapGroupManagerBean#getProperties(Properties systemConfig)	
org.rhq.core.domain.cloud.StorageNodeLoadComposite#getDiskSpacePercentageUsed()	org.rhq.core.domain.cloud.StorageNodeLoadComposite#getPartitionDiskUsedPercentage()
org.rhq.core.domain.cloud.StorageNodeLoadComposite#setDiskSpacePercentageUsed(MeasurementAggregateWithUnits diskUsedPercentage)	org.rhq.core.domain.cloud.StorageNodeLoadComposite#setPartitionDiskUsedPercentage()

Additionally, the **`org.rhq.enterprise.server.measurement.MeasurementDataManagerRemote#getAggregate`** class has been changed from **`org.rhq.core.domain.measurement.MeasurementAggregate`** to **`org.rhq.enterprise.server.measurement.MeasurementAggregate`**. This is the same class; only the package is different. Command-line scripts will be unaffected by this. For Java clients which depend on the EJB API, the clients **`import`**s should be updated and then recompiled.

2.5. Discovery Callbacks for Agent Plug-ins

A new callback has been added for the discovery components in agent plug-ins. Now, agents can intercept the information for newly-discovered resources and modify the resource data before sending the details to the server.

This new callback is implemented in the **`<discovery-callbacks>`** section in the plug-in descriptor and the **`org.rhq.core.pluginapi.inventory.ResourceDiscoveryCallback`** interface.

3. JBoss Operations Network Prerequisites and Basic Installation

The list of supported platforms for JBoss Operations Network are listed at https://access.redhat.com/site/articles/112523.

3.1. Supported Configuration Information

3.1.1. Changes in Required Java Versions

JBoss ON supports two versions of Java:

- Java 6 (deprecated)
- Java 7 (new)

Both Oracle and OpenJDK Java are supported.

The server requires the JDK, while the agent only requires the JRE.

3.1.2. Changes in Supported Web Browsers

JBoss ON has added these browsers that are used for testing and are supported for version 3.2.

- Firefox 17 ESR (new)
- Internet Explorer 9 (new)

3.2. Quick Setup for JBoss Operations Network 3.2

The *Installation Guide* has more information on configuring databases for the JBoss ON server, upgrading servers, and other installation information. This is a quick summary of performing an initial installation.

1. Set up a database named **rhq** and a database user named **rhqadmin**. Make sure the database user has adequate permissions to perform operations on the JBoss ON database.

   For more detailed database setup information:

   - Configuring PostgreSQL
   - Configuring Oracle

2. Download the JBoss ON binaries from the Customer Portal.

   a. In the Customer Portal, click the **Downloads** tab, and then the **Downloads** icon in the page.

   b. Select the **JBoss Operations Network** link under the **System Management** area in the **Downloads** page.

   c. Download the **JBoss Operations Network 3.2 Base Distribution** package by clicking the **Download** icon.

   d. There are additional plug-in packs available for EAP, EDS, EWS, and SOA-P. If any of those plug-ins will be used with the JBoss ON server, then download them as well.

3. Unzip the server distribution to the desired home directory for JBoss ON. For example:

   ```
   [root@server ~]# unzip jon-server-3.2.zip -d /opt/jon
   ```

   This creates a version-specific installation directory, **/opt/jon/jon-server-3.2**. A directory with this name should not exist prior to the unzip operation.

4. *Optional*. By default, the installer script assumes that the backend database is a PostgreSQL server running on the same system as the server. The server configuration settings — such as the database password, the server port numbers, the server name, and the way it handles database schema — use predefined defaults. One parameter, the bind address for the server, is empty and prompted by the control script if it is not set.

The parameters are set in the ***serverRoot*/jon-server-3.2/bin/rhq-server.properties** file.

5. Run the JBoss ON control script to configure the server and other services. If the **rhq-server.properties** file is not edited, then the script prompts for a bind address for the server; this can be set to **0.0.0.0**.

```
[root@server ~]# serverRoot/jon-server-3.2/bin/rhqctl install --start
23:07:00,901 INFO  [org.jboss.modules] JBoss Modules version 1.2.2.Final-
redhat-1

The [jboss.bind.address] property is required but not set in [rhq-
server.properties].
Do you want to set [jboss.bind.address] value now?
yes|no: yes
jboss.bind.address: 0.0.0.0
Is [0.0.0.0] correct?
yes|no: yes
```

This command does two things:

- Configure the JBoss ON server, a storage node, and an agent.
- Start all services when the configuration process is complete.

3.3. Upgrading to JBoss Operations Network 3.2

WARNING

It is not possible to revert your JBoss ON server to the previous version after it is upgraded. Back up all data before upgrading.

The *Installation Guide* has more information on configuring databases for the JBoss ON server, upgrading servers, and other installation information. This is a quick summary of performing an initial installation.

1. Stop the JBoss ON agent *running on the server machine*. Agents installed with the server are not autoupdated and must be updated with the JBoss ON server. All other agents will update themselves automatically when the server is upgraded.

   This is done by stopping the agent service or by using the **exit** command at the agent command prompt.

```
[root@server ~]# agentRoot/rhq-agent/bin/rhq-agent.sh
> exit
```

2. *Windows only.* If the **RHQ_AGENT_RUN_AS** or **RHQ_AGENT_RUN_AS_ME** parameter was set in the agent's **rhq-agent-env.bat** file, then there must be a password and the password prompt must be disabled.

```
RHQ_AGENT_PASSWORD=secret
RHQ_AGENT_PASSWORD_PROMPT=false
```

NOTE

If one of the **RHQ_AGENT_RUN_AS*** parameters is set without a password, then the agent upgrade process hangs.

Alternatively, the **RHQ_AGENT_RUN_AS*** parameter can be removed prior to upgrading.

3. Stop any currently running JBoss ON instances.

```
[root@server ~]# serverRoot/jon-server-3.1.0.GA/rhq-server.sh stop
```

If the JBoss ON server being upgraded is part of a high availability cloud, then all of the existing JBoss ON instances must also be stopped.

4. Download the server packages from the Customer Portal.

    a. Click **Software** in the top menu.

    b. In the **Product** field, select the **JBoss Operations Network** product.

5. Back up the server database before going through the upgrade script. In case there is a problem with the upgrade process, the backup allows you to restore to its previous state.

6. Unzip the server packages.

```
[root@server ~]# unzip jon-server-3.2.zip -d serverRoot/jon-server-3.2
```

IMPORTANT

Do not copy the new server installation on top of a previous server installation.

The directory structure within the server package gives the new server installation directory a version-specific name, such as **/opt/jon/jon-server-3.2**.

7. *Optional.* Copy over any changes in the original **rhq-server.properties** file to the new file in ***serverRoot*/jon-server-3.2/bin**. Changes to this file include things like setting up SSL and enabling SMTP for email notifications.

8. Run the **rhqctl** script with the **upgrade** subcommand and options for the original server and agent installation directories. For example:

```
[root@server ~]# ./serverRoot/jon-server-3.2/bin/rhqctl upgrade --server-dir
/opt/rhq/rhq-server-old --from-agent-dir /home/rhq/rhq-agent-old --run-data-
migrator
```

The **upgrade** command installs and configures a storage database automatically as part of upgrade; the **--run-data-migrator** option migrates the existing monitoring data into the new database. Otherwise, all historic monitoring data would be lost.

9. Start the JBoss ON server and agent.

```
[root@server ~]# serverRoot/jon-server-3.2/bin/rhqctl start
```

4. Known Issues in JBoss Operations Network

This section contains a subset of the known issues at the time that JBoss Operations Network 3.2 was released. When possible, workarounds are provided.

For a complete list of issues reported with JBoss ON 3.2, check the Bugzilla query for the JBoss ON product, with the version of 3.2.

4.1. Installation and Upgrade

Bugzilla #1025782

After upgrading an agent installed by RPM, the **rhq-agent_env.sh** file is not updated. This means that the **rhq-agent_env.sh** file — which contains the properties for the agent — contains the path to the old agent installation.

Uninventorying the old agent resource and then re-importing the agent updates the agent information.

Bugzilla #1020896

The following null pointer exception can be logged when the server is installed:

```
09:07:44,919 ERROR [org.jboss.as.ejb3.invocation] (ServerService Thread Pool
-- 67) JBAS014134: EJB Invocation failed on component SchedulerBean for
method public abstract void org.quartz.Scheduler.shutdown(boolean) throws
org.quartz.SchedulerException: javax.ejb.EJBException:
java.lang.NullPointerException
 at
org.jboss.as.ejb3.tx.CMTTxInterceptor.handleExceptionInOurTx(CMTTxIntercepto
r.java:165) [jboss-as-ejb3-7.2.1.Final-redhat-10.jar:7.2.1.Final-redhat-10]
...
Caused by: java.lang.NullPointerException
 at
org.rhq.enterprise.server.scheduler.SchedulerService.shutdown(SchedulerServi
ce.java:204) [rhq-server.jar:4.9.0.JON320ER3]
```

Bugzilla #1018238

There is an intermittent problem with installing the server, where the installation returns a JavaException and fails with the error *can't connect to 127.0.0.1:6999*.

Running the installation process again usually resolves this problem.

Bugzilla #1011254

Depending on the system configuration, installing the server on Solaris 11 fails because the installation process cannot reserve enough space for the object heap.

Resetting the heap size allows the installation process to succeed.

Bugzilla #1027339

The **rhqctl status** command does not display the correct PIDs for the server and its JVM. It displays the same PID for both processes.

On the system, the server process is named **RHQ Server** and the JVM process is called **JBossAS Java VM child process**.

4.2. Windows

Bugzilla #1004900

If a server and storage node are installed on a Windows server, an additional storage node cannot be deployed on a Linux server. The attempted storage node installation fails.

Bugzilla #960166

On Windows, there are limits on how long a pathname can be. When the JBoss ON server archive is extracted, many tools can extract it and create a directory and pathname which is longer than the limit (260 characters). When the pathname is too long, the system cannot locate the files.

Be careful when using the *extract all* command on Windows. Expanding the archive automatically creates a directory called **jon-server-***VER.RELEASE*/, which is about 20 characters. Using *extract all*, instead of specifying the directory to which to extract the archive, can double the directory name — for example, **C:\example\jon-server-3.2.0\jon-server-3.2.0** — or install it to a downloads directory such as **C:\Users\Administrator\Downloads**.

It is recommended that you extract the archive to **C:\rhq**, such as **C:\rhq\jon-server-3.2.0**.

Bugzilla #1037601

If the **JAVA_HOME** value contains parentheses, the agent cannot be installed. By default, the JRE is installed in **C:\Program Files (x86)**. An alternative location should be used for installing the agent.

Bugzilla #1037601

If **RHQ_AGENT_RUN_AS** or **RHQ_AGENT_RUN_AS_ME** parameter was set in the agent's **rhq-agent-env.bat** file is set without a password, then the agent upgrade process hangs.

If either of the **RHQ_AGENT_RUN_AS*** parameters is set, then there must be a password and the password prompt must be disabled.

```
RHQ_AGENT_PASSWORD=secret
RHQ_AGENT_PASSWORD_PROMPT=false
```

Alternatively, the **RHQ_AGENT_RUN_AS*** parameters can be removed prior to upgrading.

4.3. Agents

Bugzilla #1022979

If an agent is installed through an RPM, it cannot be managed using the **rhqctl** script. The agent RPM is intended only for agents installed on managed platforms. The **rhqctl** script, on the other hand, is to manage JBoss ON components (the server and storage nodes), so it can only be used to manage agents on a JBoss ON server system.

Bugzilla #1014631

When stopping or starting the agent using **rhqctl**, the agent log records **IllegalStateException** errors.

4.4. Server

Bugzilla #1039436

The JBoss ON server cannot be discovered and imported into the inventory. Likewise, attempting to manually import the JBoss ON server as a child of the platform fails.

Bugzilla #1035344

If some resources are being actively imported, then attempting to remove the platform resource or deleting agent fails with the error *java.lang.IllegalStateException:Failed to uninventory platform*.

Wait until the import operation completes, and then uninventory the platform. If necessary, uninventory all of the child resources for the platform, and then uninventory the platform.

Bugzilla #1029384

A JBoss ON server can be removed from the high availability cloud by removing the server on the **Servers** page in the UI. The server is successfully removed from the topology, and any agents no longer communicate with that server.

However, if the server is still installed and running on a local system, the web UI for that server can be accessed and it can perform management tasks in the JBoss ON deployment.

To prevent this, uninstall the server from the local system when it is removed from the high availability cloud.

Bugzilla #1026786

When the JBoss ON server is started, it logs a series of *JBAS015960* warnings that indicate invalid classpath entries in the deployment. These can be ignored.

Bugzilla #1019456

The server logs record a lot of warnings about not geting consistent results of the paged data:

```
14:26:52,057 WARN  [org.rhq.enterprise.server.util.CriteriaQueryRunner]
(http-/0.0.0.0:7080-6) Could not get consistent results of the paged data
and a total count for MeasurementScheduleCriteria[filter
s: [filterResourceGroupId=10021, filterResourceInventoryStatuses=
[COMMITTED]], fetches: [], sorts: []]. After 10 attempts, the collection
size is 7, while the count query reports 14 for PageControl[page
=0, size=-1, sort[definition.displayName ASC]]. The discrepancy has not
cleared up in 3750ms so we're giving up, returning inconsistent results.
Note that is most possibly NOT an error. It is likely cau
sed by concurrent database activity that changes the contents of the
database that the criteria query is querying.: java.lang.Exception
at
org.rhq.enterprise.server.util.CriteriaQueryRunner.execute(CriteriaQueryRunn
er.java:81) [rhq-server.jar:4.9.0.JON320ER3]
```

Bugzilla #1026799

If LDAP authentication is enabled, there are spurious warnings about invalid modules for each LDAP configuration property.

```
09:17:31,266 WARN  [org.jboss.security] (http-/0.0.0.0:7080-5) PBOX000234:
Invalid or misspelled module option: BindDN
09:17:31,266 WARN  [org.jboss.security] (http-/0.0.0.0:7080-5) PBOX000234:
```

```
Invalid or misspelled module option: Filter
09:17:31,266 WARN  [org.jboss.security] (http-/0.0.0.0:7080-5) PBOX000234:
Invalid or misspelled module option: java.naming.factory.initial
09:17:31,266 WARN  [org.jboss.security] (http-/0.0.0.0:7080-5) PBOX000234:
Invalid or misspelled module option: LoginProperty
09:17:31,266 WARN  [org.jboss.security] (http-/0.0.0.0:7080-5) PBOX000234:
Invalid or misspelled module option: BaseDN
09:17:31,266 WARN  [org.jboss.security] (http-/0.0.0.0:7080-5) PBOX000234:
Invalid or misspelled module option: GroupFilter
09:17:31,267 WARN  [org.jboss.security] (http-/0.0.0.0:7080-5) PBOX000234:
Invalid or misspelled module option: java.naming.provider.url
```

Bugzilla #1026836

If the backend database is stopped while a scheduled quartz job (such as a bundle deployment) is running, the quartz job cannot be rescheduled and never completes, even after the backend database is restarted and running.

4.5. Platform Plug-in

Bugzilla #1026513

File system child resources for a platform are shown as available even if they are offline. This can occur if a file system directory is unmounted or removed after being imported.

4.6. EAP 6 Plug-in

Bugzilla #1003114

EAP 6.0.0 and EAP 6.1.0 handle the keep alive value for the thread pools differently. On EAP 6.1.0, the unit is case insensitive, so the value accepts both *seconds* and *SECONDS*. On EAP 6.0.0, only all-capital letters are accepted (*SECONDS*). If a lower-case string is used, it returns an operations error, *Operation handler failed: No enum constant*.

Bugzilla #1026437

A datasource must be disabled before its configuration can be edited in the JBoss ON UI. However, the server incorrectly returns an error that a datasource is active and must be disabled when attempting to change the configuration, even if the datasource has been disabled. The configuration update is rejected.

Bugzilla #1012057

The UI allows a user to edit the resource configuration of a deployment scanner to point to a different directory name. However, if that directory does not already exist on the system, the configuration update fails (even though it appeared to succeed).

Bugzilla #1016678

If an agent discovers an EAP 6 server but the HTTP management interface definition is not included in its configuration file (such a **`host-slave.xmlM`**), then the agent incorrectly sets the management port to 0. Since the management port is not actually 0, the agent cannot connect to the EAP server.

Bugzilla #1030280

For the EAP 6 server platform MBean, the memory pool child resource metrics charts have the Y-axis improperly labeled as 00.000 rather than the real values.

Bugzilla #1033404

Updating the blocking bounded queue or bounded queue thread pool configuration properties forces a change to the value for core threads for the resource. Otherwise, the configuration update attempt fails.

Bugzilla #1014668

The XADataSource resource has an empty XA Datasource class value in the configuration page. When editing or viewing the resource configuration, it forces the user to set a value for the class or navigating away from the pages fails with the error *The following configuration properties have invalid values: [Xa Datasource Class]. The values must be corrected before the configuration can be saved.*

Bugzilla #1016698

The reload operation for the EAP 6 server fails.

Bugzilla #1013713

Whne events monitoring is enabled with a date format other than the default, the EAP 6 server becomes unavailable.

4.7. EAP 5 Plug-in

Bugzilla #1025812

Changing any value in the ExpiryQueue configuration for the JBoss Messaging resource requires a value for the expiryQueue field to be set explicitly as well, or saving the configuration fails with an exception:

```
Failed with exception: java.lang.reflect.UndeclaredThrowableException:Failed
to set property 'expiryQueue' on component
'jboss.messaging.destination:name=DLQ,service=Queue' to value
[SimpleMetaType:java.lang.String:null]. ->
javax.management.RuntimeOperationsException:null ->
java.lang.IllegalArgumentException:null object name
```

Bugzilla #1015501

Changes made to JMS queue and topic configurations are not persisted after the resources are restarted. A class cast exception is recorded in the server logs:

```
2013-10-17 06:33:06,299 WARN
[org.jboss.system.server.profileservice.persistence.deployer.ProfileServiceP
ersistenceDeployer] (main) Failed to update the persisted attachment
information
java.lang.ClassCastException:
org.jboss.system.server.profileservice.persistence.xml.PersistedSimpleValue
cannot be cast to
org.jboss.system.server.profileservice.persistence.xml.PersistedCompositeVal
ue
 at
org.jboss.system.server.profileservice.persistence.AbstractValueRecreation.c
reateMetaValue(AbstractValueRecreation.java:187)
```

Bugzilla #1019262

A datasource, queue, or topic child resource is not removed from an EAP 5 server until after the server is restarted.

4.8. EAP 4 Plug-in

Bugzilla #999494

When manually importing an EAP or EAP 5 resource into the inventory, the job is submitted and appears to complete successfully, but the EAP servers are not added to the inventory.

4.9. UI

Bugzilla #1020874

The **Add subscriptions** button the **Content->Subscriptions** tab for a platform resource returns an HTTP Status 404 error if the JBoss patch feed is selected. There are no errors recorded in the server log.

Bugzilla #959455

The number of rows shown on the **Administration->Agents** page are not lazily loaded. This means that it could display a much lower number when the page is initially loaded, and then the number increases as the user scrolls past the loaded rows.

Bugzilla #952665

When creating a new local disk storage content source in the **Administration->Content** area, click the local disk storage link returns an IllegalStateException.

Bugzilla #1029389

If a server is selected to be removed and the removal *is not confirmed* (the **No** is clicked), then all of the buttons on the **Administration->Servers** page are disabled.

Bugzilla #1020884

When uploading new content to a platform resource, clicking the **Continue** button after uploading the package returns the error *The page requested cannot be displayed due to some error*.

4.10. Provisioning

Bugzilla #1026473

Normally, the **compliance=full** setting in a bundle **deploy.xml** file instructs the provisioning process to remove any existing files and directories from the target directory and then copy in the bundle files. However, any existing *empty* directories are not removed when the bundle is provisioned.

Bugzilla #1015406

In some deployment scenarios, attempting to revert a bundle deployment results in a NullPointerException. Additionally, the revert wizard itself becomes unusable.

Bugzilla #1038597

Occasionally, there can be problems with database locks and the quartz job scheduler which prevent a bundle from being deployed successfully. These errors are recorded in the agent log.

Bugzilla #1029121

If a bundle group is assigned to a role, it cannot be deleted. Attempting to delete it fails with a ConstraintViolationException.

First remove the bundle group from any roles, and then delete it.

4.11. Monitoring

Bugzilla #1028623

When measurement data are sent to the server, the alert condition processing relies on the order to be strictly the same as generated on the agent side. Meaning, to assess the condition, no datapoint should be processed before an older (preceding) datapoint.

However, the persistence in the metrics storage node is asynchronous. This means that datapoints *can* be persisted in an order that is different than the order they were sent by the agent.

When evaluating data to process alert conditions, the datapoints are evaluated in the order in which they are persisted in the storage node, not the order in which they were created on the agent.

It is possible that this could lead to false positives to trigger an alert or to skew dampening events.

Bugzilla #1028944

For an autogroup **`Monitoring`** tab, the values displayed on the metrics tables for minimum, maximum, and average readings do not match the values charted or displayed on the individual graphs.

Bugzilla #1031820

In previous versions of JBoss ON, it was possible to override the calculated baselines for a monitoring graph and set a manual baseline. However, that ability was removed from the monitoring graphs in JBoss ON 3.2.

Bugzilla #1035890

If an alert definition is set for a trait on a resource, and the value of that trait is modified using the JBoss ON REST API, no alert is triggered.

Bugzilla #1033347

When setting a custom date range for a graph in the **`Monitoring->Metrics`** tab of a resource, the *label* for the graph displayed is for the previous date range, not custom date range.

However, the *data* displayed in the graph is properly updated to the new range.

Bugzilla #1034878

When using a custom data range, there is an intermittent error on the resource **`Summary->Activity`** page which prevents the monitoring graphs from being loaded.

5. Document History

Revision 3.2-10	**September 11, 2014**	**Ella Deon Ballard**

Fixing typos.

Revision 3.2-9	**May 24, 2014**	**Ella Deon Ballard**

Fixing typos.

Revision 3.2-8	**December 11, 2013**	**Ella Deon Ballard**

Initial release of JBoss Operations Network 3.2.